REMARKS

This response is to the Office Action dated December 21, 2008, in which claims 1-18 were rejected. Applicants respectfully request reconsideration and allowance of all pending claims in view of the following remarks.

I. REQUEST FOR TELEPHONE INTERVIEW

Applicant believes that the present claims are novel and non-obvious in view of the cited prior art, and provides below a detailed analysis of the cited art in which the differences between this art and the present claims are explained.

Applicant respectfully requests a telephone interview with the Examiner, if necessary, to avoid the possibility that the Examiner might issue a final action due to our analysis being unclear or a misinterpretation of the prior art in view of Applicant's claims.

II. CLAIM REJECTIONS UNDER §102(a,e) and §103(a)

Claims 1-2 and 5-18 were rejected under §102(a) as being allegedly anticipated by the @Risk product. Claims 3-4 were rejected as being allegedly unpatentable over the @Risk product in view of Shannon U.S. Patent No. 6,088,678.

A. **Shannon**

The Shannon patent was discussed in Applicant's prior response filed August 3, 2007, which is hereby incorporated by reference.

B **Introductory Comments**

This application relates to Risk Management software, which is designed to identify, assess and mitigate risk in a project. This requires an evaluation of a wide range of information related to the project. This information is typically recorded using Project Management software, which separates this information into 'actions', all of the actions together making up the whole project.

Risk Management software analyses the actions to create an activity breakdown structure. Using this structure, risks can then be identified and mitigating tasks determined. This can be done by the software itself, or a user can identify the risks and determine the mitigating actions and input this data into the software.

Once the mitigating tasks have been identified, the project data itself is then altered on this basis so as to reduce or prevent risk. To do this, either the existing actions in the project data are changed, or new actions are added to the project data. As a consequence of these changes, the relationships between actions can be adjusted accordingly. Hence, the Risk Management software operates independently from the Project Management software (i.e. it is 'stand-alone').

Risk Management software contrasts with risk analysis tools. These are designed to identify factors that may jeopardize the success of a project. To do this, risk analysis tools typically allow some uncertainty parameters to be added to project management data. A degree of mathematical functionality is desirable to allow some risk measurements to be evaluated on the basis of these uncertainty parameters. For example, the risk measurements might be determined using a statistical analysis of the uncertainty parameters or by use of simulations. However, the project data itself (in other words, the action information) remains unchanged as a result of the risk analysis tool.

For this reason, risk analysis tools are typically additional 'add-on' software for operation with existing Project Management software. These risk analysis tools provide additional functionality in quantifying risk using uncertainty parameters. However, they do not manage risk by adjusting the project itself to reduce risk or prevent risk from occurring.

C. The @Risk Software

@Risk is a risk analysis tool, rather than an item of Risk Management software. It is an 'add-on' for Project Management software. It therefore requires and utilizes all of the features of the Project Management software and the two operate cooperatively. In particular, @Risk includes a capability to incorporate uncertainty measurements into the Project Management software and to run project simulations that make use of these uncertainty measurements.

These capabilities are implemented using two features: probabilistic branching; and conditional modelling. Probabilistic branching allows branching from one task to any another task during a project simulation, whereas conditional modelling allows the project flow to be adjusted or project parameters to be changed on the basis of other uncertainty measurements during a simulation. This allows uncertain actions or uncertain relationships between actions in

the project to be modelled, such that a simulation can account for these different possibilities.

For example, different strategies could be employed in a project for different weather conditions, each of which may have an associated probability. A project simulation using probabilistic branching and conditional modelling could model these uncertainties in order to evaluate risk measurements. By iterative simulations, a risk measurement might be determined with a high confidence level.

However, @Risk never changes the project data itself, and these features only act as a modelling tool during simulation. Indeed, because @Risk does not change the project data, it is advantageously implemented as an 'add-on' tool for Project Management software, rather than a separate piece of software, since it can make use of the analytical functionality that the Project Management software provides.

The Examiner refers to reference V, page 1 and reference U page 3 as showing that the @Risk risk analysis tool changes the project data. However, reference U, page 3, merely refers to the generation of simulation outputs but never suggests changing the project data itself. The software just creates an uncertainty as mentioned above.

When reference V mentions "variations in activities", this just reflects that, "Variations in duration of activities are commonplace in the construction industry." (Abstract). It does not mean that the risk analysis tool alters action identifiers or creates new action identifiers in the project data of a project data store according to Applicant's claim 1.

D. **Novelty and Non-Obviousness**

The following comments relate to claim 1, although they are equally applicable to the other independent claims of this application.

Claim 1 is novel over @Risk, since @Risk lacks the feature of outputting to the project data in the project data store one or more new action identifiers or alterations to existing action identifiers in the project data, on the basis of one or more mitigating tasks identified to reduce or prevent the risk or the consequences of the risk. As explained above, @Risk does not allow the identification of mitigating tasks, leading to the project data being changed.

Moreover, @Risk also lacks the features of: accessing project data from a project data

store; analysing the project data to identify activities; and accessing changes to the project data and revising the activities. This is because @Risk is an 'add-on' tool and so integrates with Project Management software, such that it does not require these features.

It would also not be obvious for the skilled person to implement these features. Starting from @Risk, which is a risk analysis tool, the skilled person would not consider implement risk management functionality that would allow the project data to be changed, for a number of reasons.

Firstly, there is no teaching in the prior art that risk analysis tools can be turned into Risk Management software. The skilled person recognizes that a risk analysis tool provides information about risk. This analysis is made on the basis of uncertainty parameters that are related to a known project structure. However, adjusting the project would affect this analysis, thereby making the analysis less useful to the user. Hence, the skilled person would not seek to adjust the project to change the risks during risk analysis.

Moreover, the skilled person would recognize that Risk Management requires processing a large quantity of project data. With limited computational resources, this would be difficult to do within the scope of Project Management software, for example, with an 'add-on' such as @Risk. Hence, the skilled person would not seek to implement Risk Management functionality, starting from @Risk.

In fact, there are further advantages in using stand-alone Risk Management software, as defined by claim 1. Since such software can process project data separately from, and without dependence on, Project Management software, it allows the simultaneous processing of multiple projects, which may or may not interact. Moreover, it allows project data to be dynamically updated on the basis of mitigating tasks without interfering with the operation of the Project Management software. This means that the two pieces of software advantageously can operate concurrently and in tandem, without each affecting the operation of the other. This means that both the Risk Management and Project Management software can provide their required functionality more efficiently. Hence, the skilled person would not obviously arrive at the present invention, starting from @Risk.

Similar arguments can be made with respect to the other independent claims. Accordingly, Applicant respectfully requests that the rejections of claims 1-18 be withdrawn.

The Director is authorized to charge any fee deficiency required by this paper or credit any overpayment to Deposit Account No. 23-1123.

Respectfully submitted,

WESTMAN, CHAMPLIN & KELLY, P.A.

By: /David D. Brush/
David D. Brush, Reg. No. 34,557
900 Second Avenue South, Suite 1400
Minneapolis, Minnesota 55402-3319
Phone: (612) 334-3222 Fax: (612) 334-3312

DDB/